

## **REMARKS**

### **Claim Status**

Claims 1-4, 7-15, 18-25 and 29-34 are pending. Claims 1-4, 7-15, 18-25 and 29-34 are rejected.

### **Claim Rejection under 35 U.S.C. § 102**

Claims 1-4, 8, 11-15, 19, 22-25, 30, 33 and 34 are rejected under 35 U.S.C. 102(b) as being anticipated by Perlman (U.S. Publication 2002/0135696).

In the previous office action response filed on September 12, 2006, the Applicants' argued that Perlman does not disclose the limitation of *synchronizing each converted data stream to an output frame rate consistent with the first set of display attributes*, recited in independent claims 1, 12, and 23. To reiterate, Perlman does not disclose the use of any *frame rate conversion to synchronize the source material to a frame rate consistent with the display*, because Perlman requires an ad hoc correction (flicker filter) to be applied in order to reduce image flicker when displaying progressive images on an interlaced display. (Please see the previous office action response for detailed arguments.)

In response to this argument, the Examiner indicates that one skilled in the art would recognize a deinterlacer to change rate of video, and this deinterlacer is "interpreted as some sort of line doubler." The Examiner argues that, "since, in interlaced format, a frame is equal to two fields, by deinterlacing, the frame rate is doubled." The Applicants' respectfully disagree with the Examiner's assertion.

Deinterlacing does not change the *frame* rate of the video source. In this respect, it appears that the Examiner has confused the concept of "frame rate" with the concept of "field rate." On the one hand, frame rate is the number of complete frames per second (fps) in the video stream. Both interlaced and progress scan video have frame rates. On the other hand, field rate is a concept that only applies to interlaced video. When a video stream is interlaced, each frame is scanned first with odd lines, then with even lines. Such scan of every second line is called a "field." Thus, odd lines are called "odd field," and even lines are called "even field." Field rate refers to the number of odd or even fields per second in the interlaced video stream. (Please see <http://en.wikipedia.org/wiki/Interlaced>.)

For example, assume a video stream has a frame rate of 30 fps. After this video stream is interlaced into odd and even fields, it has a *field* rate of 60 fields-per-second. However, its *frame* rate does not change and remains at 30 fps.

One skilled in the art understands that deinterlacing is “the process of converting interlaced video (a sequence of fields) into a non-interlaced form (a sequence of frames).” (Please see <http://en.wikipedia.org/wiki/Deinterlacing>.) Although deinterlacing is sometimes referred to as “line doubling,” there are many different types of deinterlacing algorithms. Broadly speaking, deinterlacing methods may be categorized into two large groups: *combination*, where the even and odd frames are combined into one image and then displayed, and *extension*, where each frame (with only half the lines) is extended to the entire screen.

Combination deinterlacing algorithms include weaving (adding consecutive fields together), blending (averaging consecutive fields), selective blending, motion compensation (using motion compensation techniques to align the two fields), inverse Telecine (reversing algorithm when Telecine was used to interlace the video originally), Telecide (copying missing field from a matching previous/next frame), etc.

Frame extension deinterlacing algorithms include half-sizing (displaying each interlaced frame at half vertical resolution), line doubling (doubling the lines of each interlaced frame to fill the entire frame), etc.

Regardless of which deinterlacing algorithm is used, the *frame* rate of the video stream does not change. In the above example, the interlaced video has a 30 fps frame rate and a 60 fields-per-second field rate. With combination algorithms, after the odd and even fields are combined into a single frame, the frame rate of the deinterlaced video is still 30 fps, whereas there is no more field left. With extension algorithms, when odd field of a frame is extended, even field of the same frame is discarded, and vice versa. Again, the frame rate of the deinterlaced video remains at 30 fps. Therefore, deinterlacing process does not change the *frame* rate of the video stream being deinterlaced.

Moreover, the frame rate of a deinterlaced video stream does not necessarily match the native frame rate of a particular display device used to display the video, and therefore needs to be converted again to match the native frame rate of the particular display device.

There are many different types of display devices, both interlaced and progress scan, that have different native resolutions and frame rates. For example, current progress scan display devices typically support native frame rates of 24 fps, 25 fps, 30 fps, 50 fps, 60 fps, etc., depending on the qualities of the display device. In the above example, the deinterlaced video has a 30 fps frame rate, and this frame rate is suitable for a display device also having a native frame rate of 30 fps. But if this deinterlaced video is displayed on a device with a native frame rate of 50 or 60 fps or any frame rate other than 30 fps, it would not be suitable and the frame rate would need to be converted.

In independent claims 1, 12 and 23, the frame rate conversion synchronizes the converted data stream to an output frame rate *consistent with the first set of display attributes*. The first set of display attributes is associated with a first display device. This indicates that the output frame rate is display device driven – that is, the output frame rate is determined and converted *based on the native frame rate of the particular display device employed at the moment* and changes when the display device is changed. It is not enough that the frame rate is merely changed, but the output frame rate *must* match the native frame rate of the display device.

The Examiner further indicates that a progressive image displayed on an interlaced display is *inherently* interlaced. The Applicant respectfully disagrees. When a progressive image is displayed on an interlaced display in its native un-interlaced format, the image is distorted. This is why the image needs to be interlaced *before* it is sent to the interlaced display device. Interlacing is not done automatically whenever a progressive image is sent to an interlaced display. It must be performed as a separate step. Therefore, progressive image displayed on an interlaced display is not inherently interlaced.

For above reasons, Perlman does not disclose the frame rate conversion limitation recited in independent claims 1, 12 and 23, and independent claims 1, 12 and 23 are patentable over Perlman. Dependent claims 3-4, 8, 12-15, 19, 22, 24-25, 30, 33 and 34 depend on independent claims 1, 12 and 23 respectively, and recite additional limitations. Therefore, dependent claims 3-4, 8, 12-15, 19, 22, 24-25, 30, 33 and 34 are patentable over Perlman for at least the same reasons explained above for independent claims 1, 12 and 23.

### **Claim Rejection under 35 U.S.C. § 103**

Claims 7, 18 and 29 are rejected under 35 U.S.C. 103(a) as being unpatentable over Perlman in view of Naegle (U.S. Publication 2004/0012577).

The following is a quotation of MPEP 2142:

To establish a prima facie case of obviousness, three basic criteria must be met. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings. Second, there must be a reasonable expectation of success. *Finally, the prior art reference (or references when combined) must teach or suggest all the claim limitations* (emphasis added).

As explained above, Perlman does not disclose the limitation of *synchronizing each converted data stream to an output frame rate consistent with the first set of display attributes*, recited in independent claims 1, 12, and 23. Similarly, Naegle also does not disclose this

limitation. Therefore, Perlman and Naegle combined do not teach all the claim limitations recited in independent claims 1, 12 and 23.

Claims 7, 18 and 29 depend on independent claims 1, 12 and 23 respectively, and recite additional limitations. Therefore, Perlman and Naegle combined do not teach all the claim limitations recited in claims 7, 18 and 29. Consequently, claims 7, 18 and 29 are patentable over Perlman and Naegle.

### **CONCLUSION**

The Applicants' believe that all pending claims are allowable in view of the remarks above. Should the Examiner believe that a further telephone conference would expedite the prosecution of this application, the undersigned can be reached at the telephone number set out below.

Respectfully submitted,  
BEYER WEAVER LLP

/Michael J. Ferrazano/  
Michael J. Ferrazano  
Reg. No. 44,105

P.O. Box 70250  
Oakland, CA 94612-0250  
(408) 255-8001 Phone  
(408) 255-8002 Fax